AD-A044 904 STANFORD UNIV CALIF SYSTEMS OPTIMIZATION LAB F/G 12/1
A SUBSTITUTE INVERSE FOR THE BASIS OF A STAIRCASE STRUCTURE LIN--ETC(U)
DEC 76 R D WOLLMER N00014-75-C-0865
UNCLASSIFIED SOL-76-31 NL

| OF |
AD
A044904

END
DATE
FILMED
10-77
DDC

(11)
B.S.

A SUBSTITUTE INVERSE FOR THE BASIS OF A
STAIRCASE STRUCTURE LINEAR PROGRAM

BY

RICHARD D. WOLLMER

TECHNICAL REPORT SOL 76-31

DECEMBER 1976

Systems Optimization Laboratory

Department of Operations Research

Stanford University

Stanford California 94305

6

A SUBSTITUTE INVERSE FOR THE BASIS OF A STAIRCASE STRUCTURE LINEAR PROGRAM ,

by

10 Richard D. / Wollmer

9 TECHNICAL REPORT 14 SOL 76-31

11 Dec[redacted]1976

12 27p.

SYSTEMS OPTIMIZATION LABORATORY
DEPARTMENT OF OPERATIONS RESEARCH
Stanford University
Stanford, California

15 N00014-75-C-0865, NSF-DCR-75-04544

Research and reproduction of this report were partially supported by the Office of Naval Research Contract N00014-75-C-~~08565~~ 0865; National Science Foundation Grant DCR75-04544; and the Energy Research and Development Administration Contract EY-76-S-03-0326.

408 765

A SUBSTITUTE INVERSE FOR THE BASIS OF A

STAIRCASE STRUCTURE LINEAR PROGRAM

by

Richard D. Wollmer

1. The Staircase Structure

A staircase structured linear program is of the form:

Find $x_i \geq 0$, min z such that

$$A_1x_1 = b_1 \qquad (1.1)$$

$$B_{t-1}x_{t-1} + A_tx_t = b_t\ , \quad t = 2,\ldots,T \qquad (1.t)$$

$$c_1x_1 + c_2x_2 + \ldots + c_Tx_T = z \qquad (1.T+1)$$

where A_t is m_t by n_t, B_t is m_{t+1} by n_t, b_t is m_t by one, c_t is one by n_t, and x_t is n_t by one. The staircase linear program is represented in detached coefficient form in Table 1.

The first m_1 rows (equations) will be called period 1 rows (equations), the next m_2 rows (equations) the period 2 rows (equations), etc. Similarly, the first n_1 columns (variables) will be called the period 1 columns (variables), etc.

Dual Vector	Primal Vector Dimension	x_1	x_2	$\cdots$	x_{t-1}	x_t	x_{t+1}	$\cdots$	x_{T-1}	x_T	Right-Hand Side Constants
		n_1	n_2	$\cdots$	n_{t-1}	n_t	n_{t+1}	$\cdots$	n_{T-1}	n_T	
π_1	m_1	A_1									$= b_1$
π_2	m_2	B_1	A_2								$= b_2$
$\vdots$	$\vdots$			$\ddots$							$\vdots$
π_t	m_t				B_{t-1}	A_t					$= b_t$
π_{t+1}	m_{t+1}					B_t	A_{t+1}				$= b_{t+1}$
$\vdots$	$\vdots$							$\ddots$			
π_T	m_T								B_{T-1}	A_T	$= b_T$
minimize		C_1	C_2	$\cdots$	C_{t-1}	C_t	C_{t+1}	$\cdots$	C_{T-1}	C_T	

TABLE 1: Staircase Structure, Detached Coefficient Tableau

2. Representation of the Basis Inverse

Consider any basis of the program (1.1)-(1.T+1). Clearly any such basis must also be staircase with

$$\sum_{i=1}^{t} m_i \leq \sum_{i=1}^{t} n_i , \qquad t = 1,\dots,T \tag{2}$$

for if not the first $m_1 + \dots + m_t$ rows would be dependent. Of course equality holds in (2) for $t = T$. Let

$$a_i = \sum_{t=1}^{i} n_i - \sum_{t=1}^{i} m_i \tag{3}$$

Then the staircase structure basis may be further broken down to the form in Table 2.

Theorem 2.1. Through a suitable rearrangement of the basis, B, columns, B may be put in the form of Table 2 such that the submatrix consisting of the first $\Sigma_{i=1}^{t} m_i$ rows of columns of B is nonsingular, $t = 1,\dots,T$.

Proof: Clearly the first $m_1 + m_2 + \dots + m_{t-1}$ columns in this restriction of B are independent for if not B is singular. Similarly any basis for the first $m_1 + \dots + m_t$ rows of B must consist of $m_1 + \dots + m_t$ columns. Consequently there is a subset of the columns of the first $m_1 + \dots + m_T$ rows of B consisting of the first $n_1 + \dots + n_{t-1}$ columns and $m_t - a_{t-1}$ of the next m_t columns that are independent. Hence the theorem is proven as one may rearrange the first n_1 columns for $t = 1$, the next n_2 columns for $t = 2$, etc.

Dual Vector	Primal Vector Dimension	x_{11} m_1	x_{12} a_1	x_{21} m_2-a_1	x_{22} a_2	... ...	$x_{t-1\ 1}$ $m_{t-1}-a_{t-2}$	$x_{t-1\ 2}$ a_{t-1}	x_{t1} m_t-a_{t-1}	x_{t2} a_t	... ...	$x_{T-1\ 1}$ $m_{T-1}-a_{T-2}$	$x_{T-1\ 2}$ a_{T-1}	x_T m_T-a_T	Right Hand Side Constants
π_1	m_1	A_{11}	A_{12}												$= b_1$
π_2	m_2	B_{11}	B_{12}	A_{21}	A_{22}										$= b_2$
π_3	m_3			B_{21}	B_{22}										$= b_3$
$\vdots$	$\vdots$					$\ddots$									$\vdots$
π_t	m_t						$B_{t-1\ 1}$	$B_{t-1\ 2}$	A_{t1}	A_{t2}					$= b_t$
π_{t+1}	m_{t+1}								B_{t1}	B_{t2}					$= b_{t+1}$
$\vdots$	$\vdots$										$\ddots$				
π_{T-1}	m_{T-1}											$A_{T-1\ 1}$	$A_{T-1\ 2}$		$= b_{T-1}$
π_T	m_T											$B_{T-1\ 1}$	$B_{T-1\ 2}$	A_T	$= b_T$
minimize		C_{11}	C_{12}	C_{21}	C_{22}		$C_{t-1\ 1}$	$C_{t-1\ 2}$	C_{t1}	C_{t2}	...	$C_{T-1\ 1}$	$C_{T-1\ 2}$	C_T	

TABLE 2: Breakdown of Staircase Structure Basis, Detached Coefficient Form

We will assume throughout that the columns of B are rearranged so as to satisfy the conditions of Theorem 2.1. Then in Theorem 2.2 it will be shown that through a series of elementary operations B may be placed in the form of Table 3.

Theorem 2.2. The staircase structure basis may be reduced to the form displayed in Table 3 with

$$K_1 = A_{11}^{-1} \tag{4.1a}$$

$$\bar{b}_1 = K_1 b_1 \tag{4.1b}$$

$$\bar{A}_{t2} = K_t[A_{t2}] \tag{4.tc}$$

$$K_t = [B_{t-1\,2} - [0|B_{t-1\,1}]\,\bar{A}_{t-1\,2}|A_{t1}],^{-1} \qquad t > 1 \tag{4.ta}$$

$$\bar{b}_t = K_t[b_t - [0|B_{t-1\,1}]\,\bar{b}_{t-1}], \qquad t > 1 \tag{4.tb}$$

where in (4.tb-c), 0 is a column of a_{t-2} zero vectors and $a_0 = 0$.

Proof: By Theorem 2.1, $K_1 = A_{11}^{-1}$ is well defined and multiplying by K_1 yields the desired form for the first m_1 equations. Suppose the theorem holds for equations $m_1, m_2, \ldots, m_{k-1}$, and the basis B is placed in the form of Table 3 for these equations. Subtracting $[0|B_{k-1\,1}]$ times the $k-1$ period rows from the k period rows yields $[B_{k-1\,2} - [0|B_{k-1\,1}]$ $\bar{A}_{k-1\,2}|A_{k1}A_{k2}][{x_{k-1\,2} \atop x_k}] = b_k - [0|B_{k-1}]\bar{b}_{k-1}$. By Theorem 2.1 the first m_k columns of the above coefficients must be independent and $K_k = [B_{k-1\,2} - [0|B_{k-1\,1}]\bar{A}_{k-1\,2}|A_{k1}]^{-1}$ exists and clearly the theorem holds for $t = 1,\ldots,k$. The theorem then follows from induction.

Primal Vector Dimension	x_{11}	x_{12}	x_{21}	x_{22}	x_{31}	$\cdots$	$x_{T-2\ 2}$	$x_{T-1\ 1}$	$x_{T-1\ 2}$	x_T	Right-Hand Side Constants
	m_1	a_1	m_2-a_1	a_2	m_3-a_2	$\cdots$	a_{T-2}	$m_{T-1}-a_{T-2}$	a_{T-1}	m_T-a_{T-1}	
m_1	I	$\bar{A}_{12}$									$= \bar{b}_1$
m_2				$\bar{A}_{22}$							$= \bar{b}_2$
m_3						$\cdot$					$= \bar{b}_3$
$\cdot$						$\cdot$					$\cdot$
$\cdot$						$\cdot$					$\cdot$
m_{T-2}							$\bar{A}_{T-2\ 2}$				$= \bar{b}_{T-2}$
m_{T-1}									$\bar{A}_{T-1\ 2}$		$= \bar{b}_{T-1}$
m_T											$= \bar{b}_T$

TABLE 3: Reduced Form of Staircase Structure Basis

From this, $x_1,\ldots,x_T$ may be found as follows

$$[x_{T-1\ 2}\ x_T] = \bar{b}_T = \bar{b}'_T \qquad (5.T)$$

$$[x_{T-2\ 2}\ x_{T-1\ 1}] = \bar{b}_{T-1} - \bar{A}_{T-1\ 2}\, x_{T-1\ 2} = b'_{T-1} \qquad (5.T-1)$$

$$\vdots$$

$$[x_{12}\ x_{21}] = \bar{b}_2 - \bar{A}_{22}x_{22} = \bar{b}'_2 \qquad (5.2)$$

$$[x_{11}] = \bar{b}_1 - \bar{A}_{12}x_{12} = \bar{b}'_1 \qquad (5.1)$$

Note that in this form a period t variable may be basic for period t or for period $t+1$ only.

[illegible]ting the Substitute Inverse

Suppose a new variable is to enter the basis. Let t be the period of the entering variable and k the period of the exiting variable. Then there are three cases to consider: $k = t$, $k > t$, or $k < t$. Call x_{tj} the entering variable and x_{ki} the exiting variable.

Case 1. $k = t$. There are two subcases to consider: the exiting variable x_{ki} is basic for period t or the exiting variable is basic for period $t+1$. In either situation, none of $a_1,\ldots,a_{T-1}$ will change according to (3). Also $K_1,\ldots,K_{t-1}$ will not change. Two methods for updating $K_t,\ldots,K_T$ will be presented.

Pivoting Method. To form the new K_t if the exiting variable is basic for period t, one simply multiplies the period t coefficient vector of x_{tj} by the current K_t, append it to K_t and pivot on the element in that column belonging to the row for which the exiting variable is basic. Then K_s for $s = t+1,\ldots,T$ are found recursively as follows. Compute the new $\bar{A}_{s-1\,2} = K_{s-1}A_{s-1\,2}$. Consider the following matrix

$$K_s^{old} | K_s^{old}[B_{s-1\,2} - [0|B_{s-1\,1}]\bar{A}^{new}_{s-1\,2} | A_{s1}] = K_s^{old} | K_s^{old}[\bar{B}_{s-1\,2} | A_{s\,1}] \quad (6)$$

Clearly the last $m_s - a_{s-1}$ columns form the last $m_s - a_{s-1}$ columns of the identity matrix. Hence the new K_s is obtained from the old one by appending to it the matrix $K_s^{old}\bar{B}_{s-1\,2}$, pivoting on the first a_{s-1} rows of the appended columns, and then dropping them.

For the situation where the exiting variable is basic for period $t+1$, K_t will remain unchanged. However, A_{t2} and B_{t2} will change with column x_{tj} replacing column $x_{ki} = x_{ti}$. Then $K_{t+1},\ldots,K_t$ are obtained as in the previous situation with the new K_{t+1} being obtained from the old one by a single pivot and the new K_s obtained from the old by a_{s-1} pivots for $s=t+2,\ldots,T$.

<u>Dyad Matrix Method</u>.* If B is an m by m non-singular matrix, C is an m by 1 column vector, and R a 1 by m row vector, then

$$[B + CR]^{-1} = [B^{-1} + k\bar{C}\bar{R}] \quad (7.1)$$

*This approach was suggested to the author by George B. Dantzig.

where

$$\bar{C} = B^{-1}C \tag{7.2}$$

$$\bar{R} = RB^{-1} \tag{7.3}$$

$$k = \frac{-1}{1 + RB^{-1}C} \quad . \tag{7.4}$$

The reader may verify the above by multiplying $B+CR$ by the right hand side of (7.1). The matrix CR, which is the product of a column and a row vector, is called a dyad matrix. Hence if one adds a dyad matrix to B, the inverse of the resulting matrix is obtained by adding a dyad matrix to B^{-1}.

From (7) it follows recursively that if R_i and C_i are column vectors then

$$\left[B + \sum_{i=1}^{n} C_i R_i\right]^{-1} = \left[B^{-1} + \sum_{i=1}^{n} k_i \bar{C}_i \bar{R}_i\right] \tag{8.1}$$

where

$$\bar{C}_i = B_i^{-1} C_i \tag{8.2}$$

$$\bar{R}_i = R_i B_i^{-1} \tag{8.3}$$

$$k_i = \frac{-1}{1 + R_i B_i^{-1} C_i} \tag{8.4}$$

$$B_i^{-1} = \left[B_i^{-1} + \sum_{j=1}^{i-1} k_i \bar{C}_i \bar{R}_i\right] \quad i \geq 2. \tag{8.5}$$

Hence if one adds n dyad matrices to B, one adds n dyad matrices to its inverse. Note that it follows from induction in the above that each of the $\bar{R}_i$ is the product of a row vector and B^{-1}. Note also that this representation is not unique and in fact in (8) depends on the ordering of the dyad matrices.

Performing a pivot operation is equivalent to adding a dyad matrix. Specifically, R is the pivot row and C is such that its i^{th} component is the multiple of R to be added to row i. Also replacing column P_j by P_j^* in a matrix is equivalent to adding a dyad matrix with $C = P_j^* - P_j$ and R is the unit vector with a one in position j.

<u>Theorem 3.1</u>. Suppose $K_s^{new} = K_s^{old} + \sum_1^{\ell} C_i R_i K_s^{old}$ where C_i and R_i are column and row vectors respectively. If B_{s1}, A_{s2}, B_{s2}, and A_{s+11} are unchanged then $K_{s+1}^{new} = K_{s+1}^{old} + \sum_1^{\ell} C_i' R_i' K_{s+1}^{old}$ where C_i' and R_i' are column and row vectors respectively.

<u>Proof</u>: It follows from (4.tc) that $\bar{A}_{s2}^{new} = \bar{A}_{s2}^{old} + \sum_1^{\ell} C_i R_i K_s^{old} A_{s2}$ and consequently from (4.ta) that $[K_{s+1}^{new}]^{-1} = [K_{s+1}^{old}]^{-1} + \sum_1^{\ell} \bar{C}_i \bar{R}_i$ where $\bar{R}_i = R_i K_s^{old} A_{s2}$ and $\bar{C}_i = -[0|B_{s1}]C_i$. The theorem follows from (8).

<u>Theorem 3.2</u>. Suppose $K_s^{new} = K' + CRK_s^{old}$, and $[0|B_{s1}^{new}] = [0|B_{s1}^{old}] + C'R$ where $K' = K_s^{old} + \sum_{i=1}^{\ell-1} C_i R_i K_s^{old}$; C, C_i and C' are column vectors, R_i are row vectors and R is a unit row vector. Suppose also B_{s2} and A_{s+11} are unchanged. Then $K_{s+1}^{new} = K_{s+1}^{old} + \sum_{i=1}^{\ell} \bar{C}_i \bar{R}_i K_{s+1}^{old}$ where the $\bar{C}_i$ are column vectors and the $\bar{R}_i$ are row vectors.

Proof: From (4.ta) and (4.tc) it follows that $[K_{s+1}^{new}]^{-1} - [K_{s+1}^{old}]^{-1}$ $= [-MA_{s2}|0]$ where $M = [[0|B_{s1}^{old}] + C'R][K'+CRK_s^{old}] - [0|B_{s1}^{old}]K_s^{old}$ or $M = [[0|B_{s1}^{old}] + C'R][K'-K_s^{old}] + [C'+[0|B_{s1}^{old}]C + C'RC]RK_s^{old}$. Thus M and consequently $-MA_{s2}$ is the sum of ℓ dyad matrices. The theorem follows from repeated application of (7.1) and (7.3).

Now consider what happens when the entering and exiting basis variables are both from period t. If they are basic for period $t+1$ then $K_1,\ldots,K_t$ are unchanged and K_{t+1}^{new} is obtained from K_{t+1}^{old} by a single pivot operation which is equivalent to adding a dyad matrix. From Theorem 3.1 it follows that K_s^{new} is obtained from K_s^{old} by adding a single dyad matrix whose rows are linear combinations of the rows of K_s^{old} for $s > t+1$.

If the entering and exiting variable are basic for period t, then $K_1,\ldots,K_{t-1}$ are unchanged and K_s is updated by adding a single dyad matrix whose rows are linear combinations of the rows of K_s^{old} for $s \geq t$. For $s = t+1$, this follows from Theorem 3.2 and for $s > t+1$ this follows from Theorem 3.1.

Case 2. $k > t$. For this case, each of $a_t,\ldots,a_{k-1}$ increase by one while all other a_i remain unchanged according to (3). Suppose x_{ki} is basic for period k. Let x_{tj} be basic for period $t+1$. Then a variable $x_{t+1\ j(t+1)}$ currently basic for period $t+1$ must become basic for period $t+2$ instead, and continue in this manner until finally a variable $x_{k-1\ j(k-1)}$ basic for period $k-1$ becomes basic for period k replacing the exiting variable x_{kj}. If x_{kj} is basic for period $k+1$ the replacement process continues for one more period. For this case $K_1, \ldots, K_t$ are unchanged.

Pivoting Method. For K_{t+1} consider (6) for $s = t+1$ and where $\bar{B}_{t2}$ is appended by one column whose coefficients are generated by the coefficients of the entering variable x_{tj}. Pivot on any nonzero element of column x_{tj} in a position a_t+1 or higher. (There must be at least one such element or Theorem 2.1 is contradicted.) Let ℓ be the pivot row. Then the ℓ^{th} column of the old period $t+1$ basis (which must be a column of $A_{t+1\ 1}$) is deleted from the period $t+1$ basis and introduced in the period $t+2$ basis. The process continues for $K_{t+2}, \ldots, K_k$. Thus for K_s, $t+2 \leq s \leq k-1$, $A_{s-1\ 2}$ is appended by the column deleted from $A_{s-1\ 1}$. Then K_s is obtained by pivoting on the first $a_{s-1} + 1$ rows of the m_s by $a_{s-1} + 1$ (i.e., a_{s-1} is increased by one) matrix $K_s^{old}\bar{B}_{s-1\ 2}$. The variable deleted from the period s basis is the one which has a one in the same position as that of the column introduced in the period s basis. That variable is then introduced into the period $s+1$ basis. For $s = k$ the procedure is the same except the variable leaving the period k basis must be x_{ki} and it of course is not introduced into any other basis. For $s = k+1, \ldots, T$ the computation of K_s is identical to that of Case 1. If x_{ki} is basic for period $k+1$, it leaves the period $k+1$ basis instead of the period k basis and the procedure is the same except for minor changes in the sequence of values for s.

Dyad Matrix Method. Here, in many cases the A_{s2}, B_{s2}, and A_{s+11} will not always remain the same. However, as will be seen later on, when this happens, only one column of the matrix $[B_{s2} A_{s+11}]$ changes. This can occur by replacing a column of B_{s2} (and consequently a column of A_{s2}) by another column in B_{s2}, deleting a column from B_{s2} and adding a column to A_{s+11}, etc. Such an operation will be called an elementary column replacement.

Theorem 3.3. Suppose $K_s^{new} = K_s^{old} + \sum_1^{\ell} C_i R_i K_s^{old}$ where C_i and R_i are column and row vectors respectively. Then $K_{s+1}^{new} = K_{s+1}^{old} + \sum_{i=1}^{\ell+1} C_i' R_i' K_{s+1}^{old}$ (and possibly a permutation of the columns) if $[B_{s2}^{new} A_{s+1}^{new}]$ is obtained from $[B_{s2}^{old} A_{s+1}^{old}]$ by an elementary column replacement and where C_i' and R_i' are column and row vectors respectively.

Proof: Case 1: A_{s+1}^{new} is obtained by a single column replacement in A_{s+1}^{old}. From Theorem 3.1, replacing $\bar{A}_{s2}^{old}$ with $\bar{A}_{s2}^{new}$ is equivalent to adding ℓ dyad matrices to $[K_{s+1}^{old}]^{-1}$ in (6). Replacing A_{s+11}^{old} with A_{s+1}^{new} is a single column replacement adding another dyad matrix to $[K_{s+1}^{old}]^{-1}$ to obtain $[K_{s+1}^{new}]^{-1}$ and the theorem follows for this case. Case 2: A column of B_{s2} (and consequently A_{s2}) is replaced. Replacing K_s^{old} with K_s^{new} in (4.5c) and (6) again adds ℓ dyad matrices to $[K_{s+1}^{old}]^{-1}$. Then replacing A_{s2}^{old} and B_{s2}^{old} with A_{s2}^{new} and B_{s2}^{new} is equivalent to a simple column replacement thus adding another dyad matrix to obtain $[K_{s+1}^{new}]^{-1}$. The theorem follows for this case. Case 3: A column of B_{s2} (and A_{s2}) is dropped and an additional column is appended to A_{s+11}. This is equivalent to replacing the outgoing column of B_{s2} with the incoming column of $A_{s+1\,1}$ and the outgoing column of A_{s2} with a column of zeros. Hence this case follows from case 2. Case 4: An additional column of B_{s2} (and A_{s2}) is appended and a column of A_{s+11} is dropped. Delete the outgoing column of A_{s+11} and append it to B_{s2} and append a column of zeros to A_{s2}. This reduces case 4 to case 2 and the theorem is proven.

Suppose x_k is basic for period k. $K_1,\ldots,K_t$ are unchanged. K_{t+1} is updated by a single pivot operation hence adding a single dyad matrix. Assume K_{s-1} is updated by adding ℓ dyad matrices. For $s = t+2,\ldots,k-1$ a variable basic for period $s-1$ becomes basic for period s and a variable basic for period s becomes basic for period $s+1$ thereby shifting a column from B_{s-11} (and A_{s-11}) to B_{s-12} (and A_{s-12}) and also shifting a column of B_{s1} (and A_{s1}) to B_{s2} (and A_{s2}). For updating K_s one may update the inverse by first adding a column to B_{s-12} (and A_{s-12}) and deleting a column from A_{s1} and then deleting a column of B_{s-11} and replacing it with zeros in $[0|B_{s-11}]$. By Theorems 3.2 and 3.3 K_s^{-1} and K_s would be updated by adding $\ell+1$ dyad matrices. Note that for Theorem 3.2 to apply the dyad matrix corresponding to the deletion of a column of A_{s-11} in K_{s-1}^{-1} must be acted upon first (i.e., correspond to $i=1$ in (8)). For $s=k$ the same result holds except that the column of A_k and B_k dropped is that of the exiting variable and hence is not shifted to the $k+1$ period basis. For period $k+1$, the column of B_{k1} corresponding to the exiting variable is dropped while B_{k2}, B_{k+11} and A_{k+11} are unchanged, resulting in one additional dyad matrix to be added. Hence for $s = t+2,\ldots,k$ K_s is updated by adding $s-t$ dyad matrices and for $s \geq k+1$, K_s is updated by adding $k+1-t$ dyad matrices. The result for $s \geq k+2$ follows from Theorem 3.1.

If x_k is basic for period $k+1$, the replacement goes on for an additional period but the formulas remain the same.

Case 3. $k < t$. For this case $a_k,\ldots,a_{t-1}$ each decrease by one according to (3). Suppose x_{ki} is basic for period k (k+1). Since x_{kj} leaves the period k (k+1) basis, a period k (k+1) variable in the period k+1 (k+2) basis must leave that basis and enter the period k (k+1) basis. The process continues until finally x_{tj} enters the period t basis.

Pivoting Method. Suppose x_{ki} is basic for period k. Then $K_1,\ldots,K_{k-1}$ are unchanged. The choice for the period k variable to enter the period k basis may be any variable such that K_k times its period k column coefficients yields a nonzero element in the j^{th} position. For $s = k+1,\ldots,t-1$, K_s is computed as follows. Form (6) and perform a sequence of pivots on the m_s by $a_{s-1}-1$ matrix $\bar{B}_{s-1\,2}$ to obtain $\bar{K}_s$. Then $\bar{K}_s[\bar{B}_{s-1\,2}|A_{s\,1}]$ will be the identity matrix with one column missing. Append A_{s1} with the coefficients of any period s variable, $x_{s\ell}$ currently basic for period s+1 such that $\bar{K}_s$ times its period s column coefficients yield a nonzero element in the row for which $\bar{K}_s$ has all zeros, and pivot on that element in $\bar{K}_s$ times its column coefficients. Now $x_{s\ell}$ is in the period s basis and has left the period s+1 basis. For period t, simply let the period t coefficients of X_{ti} replace the coefficients of the period t variable which has left the period t basis and pivot on the m_T by a_{T-1} matrix $K_t^{old-}B_{t-1\,2}$ to obtain the new K_t. The procedure for finding $K_{t+1},\ldots,K_T$ is then identical to that of Case 1.

If $x_{k\ell}$ is basic for period $k+1$ instead of period k the procedure is the same except that K_k is unchanged and x_{ki} leaves the period $k+1$ basis.

Dyad Matrix Method. Suppose x_{kj} is basic for period k. Then a column of A_{k2} and B_{k2} is shifted to A_{k1} and B_{k1} to replace the column coefficients of x_{kj}. A single pivot, or the addition of a single dyad matrix then updates K_k. For $s = k+1,\ldots,t-1$ an additional column is added to B_{s-11} and a column is added to A_{s+11} and B_{s+11} and deleted from A_{s+12} and B_{s+12}. Hence as in case 2, if K_{s-1} is updated by adding ℓ dyad matrices, K_s is updated by adding $\ell+1$ dyad matrices by Theorems 3.2 and 3.3. The same is true for $s=t$ except that the column added to A_{t1} and B_{t1} is x_{ti} and hence no column is deleted from A_{t2} and B_{t2}. Hence $K_1,\ldots,K_{k-1}$ are unchanged. K_s is updated by adding $s-k+1$ dyad matrices for $s = k,\ldots,t$ and by adding $t-k+2$ dyad matrices for $s \geq t+1$. The latter follows inductively from Theorem 3.3 for $s=t+1$ and Theorem 3.1 for $s \geq t+2$.

If x_{kj} is basic for period $k+1$, $K_1,\ldots,k_k$ are unchanged. An argument similar to the above shows that K_s is updated by adding $s-k$ dyad matrices for $s=k+1,\ldots,t$ and by adding $t-k+1$ dyad matrices for $s \geq t+1$.

4. Finding the Multipliers

In Theorem 4.1 it will be shown that the simplex multipliers may be found by the following recursive relationships:

$$\pi_T = \bar{C}_{T-1\ 2}\ \bar{C}_T \tag{9.T}$$

$$\pi_t = \bar{C}_{t-1\ 2}\ \bar{C}_{t1} - \pi_{t+1}[0|B_{t1}]K_t, \qquad t < T \tag{9.t}$$

where

$$\bar{C}_{11} = C_{11}K_1 \tag{10.1}$$

$$\bar{C}_{t-1\ 2}\ \bar{C}_{t1} = [C_{t-1\ 2} - [\bar{C}_{t-2\ 2}\ \bar{C}_{t-1\ 1}]A_{t-1\ 2}|C_t]K_t, \quad t > 1 \tag{10.t}$$

with $\bar{C}_{T1} = \bar{C}_T$ and $\bar{C}_{02}$ being vacuous.

Theorem 4.1. The simplex multipliers are given by (9).

Proof: From Table 2, $\pi_1 A_{11} + \pi_2 B_{11} = C_{11}$ and thus from (4.1a) $\pi_1 = C_{11}K_1 - \pi_2 B_{11}K_1$. Hence the theorem holds for $t = 1$. Suppose the theorem holds for $t = k$. For $t = k+1$ one has (i) $\pi_k A_{k2} + \pi_{k+1}B_{k2} = C_{k2}$ and (ii) $\pi_{k+1}A_{k+1\ 1} + \pi_{k+2}\ B_{k+1\ 1} = C_{k+1\ 1}$, using the convention for $k+1 = T$ that $\pi_{T+1} = 0$, $B_{T1} = |0|$ and $C_{T1} = C_T$. By the inductive assumption and (4kc), $\pi_k A_{k2} = [\bar{C}_{k-1\ 2}\ \bar{C}_{k1}]A_{k2} - \pi_{k+1}[0|B_{k1}]\bar{A}_{k2}$ and (i) may be replaced with (iii) $\pi_{k+1}[B_{k2} - [0|B_{k1}]\bar{A}_{k2}] = C_{k2} - [\bar{C}_{k-1\ 2}\ \bar{C}_{k1}]A_{k2}$. Combining (ii) and (iii) yields $\pi_{k+1}[B_{k2} - [0|B_{k1}]\bar{A}_{k2}|A_{k+1\ 1}] =$ $[C_{k2} - [\bar{C}_{k-1\ 2}\ \bar{C}_{k1}]A_{k2}|C_{k+1\ 1}] - \pi_{k+2}[0|B_{k+1\ 1}]$. By (4ka), multiplying both sides on the right by K_{k+1} and noting that $\bar{C}_{02}$ is vacuous yields (9.t) for $t = k+1$. The theorem follows from induction.

Once the simplex multipliers are known, one can find the new variable to enter the basis.

5. Finding the Exiting Variable

As in the previous section let x_{tj} be the entering variable. Then let A_{tj} and B_{tj} be its original nonzero coefficients as given in Table 1 and $\bar{a}_k$ its period k coefficients in the reduced form of Table 3. From (4) it follows that

$$\begin{aligned} \bar{a}_k &= 0, \ k < t \\ \bar{a}_t &= K_t A_{tj} \\ \bar{a}_{t+1} &= K_{t+1}[B_{tj} - [0|B_{t1}]\bar{a}_t] \\ \bar{a}_k &= -K_k[0|B_{k-1\ 1}]\bar{a}_{k-1}, \ k > T \end{aligned} \qquad (11)$$

Letting all nonbasic variables other than x_{tj} be zero it follows from (5) and (11) that

$$x_{k-1\ 2}\ x_{k1} = \bar{b}'_k - \bar{a}'_k x_{k2} \qquad (12.ka)$$

where

$$\bar{a}'_T = \bar{a}_T \qquad (12.Tb)$$

$$\bar{a}'_k = \bar{a}_k - [\bar{A}_{k2}|0]\bar{a}'_{k+1} \quad \text{for} \quad k < T \qquad (12.kb)$$

Hence the pivot row must satisfy

$$\frac{\bar{b}'_{ki}}{\bar{a}'_{ki}} = \min_{\bar{a}'_{ki} > 0} \frac{\bar{b}'_{ki}}{\bar{a}'_{ki}} \tag{13}$$

where $\bar{b}'_{ki}$ and $\bar{a}'_{ki}$ represent the i^{th} components of $\bar{b}'_k$ and $\bar{a}'_k$ respectively.

6. Concluding Remarks and Example

The development of special purpose algorithms to take advantage of staircase structures in linear programs has previously been dealt with by Dantzig [2], Ho and Manne [6], Ho [7], and Glassey [5].

Dantzig [2] uses a substitute basis as opposed to a substitute inverse for solving the more general staircase problem in which nonzero elements may appear anywhere below the diagonal.

In the decomposition of Ho and Manne [6] it was reported that computer storage requirements were greatly reduced when compared to the simplex method but computation times were significantly increased. However, Ho [7] has subsequently incorporated new techniques which reduced computation times below that of the simplex method.

Storage requirements of the method of this report should be competitive with the [6]. Computationally more frequent but faster matrix reinversions would be required using this method than with the simplex method.

If pivoting is used, then computational efficiency depends largely on the column dimensions of the a_i in Table 3. The simplex method using product form of the inverse requires multiplication by an m by m matrix which is the identity except for one column in order to update. For the substitute inverse, no updating occurs for periods earlier than the periods of both the entering and exiting variables. To update K_s one multiplies by an m_s by m_s matrix which is identical to the identity matrix except for one column if s is the first period to be updated and multiplies by a_{s-1} such matrices otherwise. Hence efficiency depends to a large extent on the a_i. Previous experience on problems of this type indicate that the a_i are usually quite small.

If the dyad matrix method is used, computational efficiency depends largely on the absolute difference between the periods of the entering and exiting variable. In particular if the entering and exiting variable are of the same period, the K_s which are updated are done so by adding one dyad matrix. Otherwise one updates by adding anywhere from one to the absolute difference between the periods of the entering and exiting variables. The addition of a dyad matrix is about the equivalent of a single pivot in terms of computation time.

Note that with either method one would want to reinvert more often in the higher numbered periods than in the lower numbered ones. If all periods had the same number of equations, one could reinvert the K_s matrices approximately T times as often as in the simplex method and still be competitive. It should also be noted that the number of dyad matrices to be added to the K_t cannot exceed T at any iteration and that this can occur only in period T and even then only if the pair of variables entering and leaving the basis are from periods 1 and T. Hence if one were to reinvert after a fixed number, say k, of dyad matrix additions to any K_s one would reinvert fewer than T times as often as one would by reinverting after k iterations of the simplex method. Of course, using this criteria, some of the K_i (in particular K_1) would be updated fewer than every k iterations and others (in particular K_T) would be updated more often than every k iterations.

The relevant parameters discussed above were gathered from a standard simplex method run of the 9 period PILOT energy model and the results are summarized in Table 4. The values of the a_i are large enough to indicate that the pivoting method would not be efficient except for period 9 and of course the first period to be updated. However, the dyad matrix method of updating appears to be extremely efficient. Noting that period 9 has much fewer constraints than periods 1 through 8, the dyad matrix update method should be competitive with the simplex method if no more than 8 dyad matrices are added per period per iteration. Table 4 shows that the last period, which is the one requiring the largest number of dyad matrix additions, requires but 3.218 per iteration. The average number of dyad matrix additions per iteration for all periods is 1.778.

Acknowledgement

The author is very much indebted to George Dantzig and Muhamed Aganagic for their helpful suggestions.

Period Number (i)	Number of Constraints	Number of Variable	Average a_i	Number of Dyad Matrix Additions* Total	Number of Dyad Matrix Additions* Average per Iteration
1	106	345	26	17	.137
2	107	346	37	53	.487
3	107	346	29	106	.895
4	107	346	31	165	1.331
5	107	346	28	229	1.847
6	107	346	29	284	2.290
7	107	346	26	339	2.734
8	124	363	2	380	3.065
9	46	46	--	399	3.218

TABLE 4: Statistics for the PILOT energy model run of 124 iterations

*These numbers are based on the assumption that the entering and exiting variables are basic to the period for which they belong (except for the entering variable in Case 2 of Section 3). The fact that this would not always be the case means actual numbers would be lower than those in the last two columns of Table 4.

References

1. G. B. Dantzig, Linear Programming and Extensions (Princeton University Press, Princeton, N.J., 1963).

2. G. B. Dantzig, "Upper Bounds, Secondary Constraints, and Block Triangularity In Linear Programming," Econometrica 23 (1965), 174-183.

3. G. B. Dantzig and P. Wolfe, "The Decomposition Algorithm for Linear Programs," Econometrica 29 (1961), 767-778.

4. C. R. Glassey, "Nested Decomposition and Multi-stage Linear Programs," Management Science 20 (1973), 282-292.

5. C. R. Glassey, "Dynamic Linear Programs for Production Scheduling, Journal of the Operations Research Society of America 19 (1971), 45-56.

6. J. K. Ho and A. S. Manne, "Nested Decomposition for Dynamic Models," Mathematical Programming 6 (1974), 121-140.

7. J. K. Ho, "Nested Decomposition of Large Scale Linear Programs With the Staircase Structure," Technical Report 74-4, Systems Optimization Laboratory, Department of Operations Research, Stanford University (1974).

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: SOL 76-31	2. GOVT ACCESSION NO.
	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): A SUBSTITUTE INVERSE FOR THE BASIS OF A STAIRCASE STURCTURE LINEAR PROGRAM	5. TYPE OF REPORT & PERIOD COVERED: Technical Report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Richard D. WOLLMER	8. CONTRACT OR GRANT NUMBER(s): N00014-75-C-0865
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Department of Operations Research - SOL, Stanford University, Stanford, CA 94305	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: NR-047-143
11. CONTROLLING OFFICE NAME AND ADDRESS: Operations Research Program, Code 434, Office of Naval Research, Arlington, Virginia 22217	12. REPORT DATE: December 1976
	13. NUMBER OF PAGES: 23
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report)
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

This document has been approved for public release and sale;
its distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ENERGY
LINEAR PROGRAMMING
LARGE SCALE SYSTEMS
STAIRCASE STRUCTURE

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

SEE ATTACHED

over

TR SOL 76-31: "A Substitute Inverse for the Basis of a Staircase Structure Linear Program", by Richard D. Wollmer

This report concerns a method for computing a substitute inverse for linear programs with a staircase structure. The constraints of a staircase structure linear program are of the following form:

$$A_1 x_1 = d_1 ;$$

$$B_{t-1} x_{t-1} + A_t x_t = d_t ; \qquad (t = 2, \ldots, T) .$$

Letting m_i be the number of constraints in period i, the substitute inverse consists of the inverse of T matrices which are $m_i \times m_i$, $i = 1, \ldots, T$ as opposed to the actual inverse which is $m \times m$, $m = \Sigma m_i$. Hence the substitute (sum over i) inverse would require significantly less storage than the actual inverse.

Two methods are presented for updating the substitute inverse, both of which consist of updating some, but not necessarily all of the T smaller inverses. The first, a pivoting method, requires appending one or more columns to the smaller inverses and pivoting on the appended columns. The second requires adding one to T dyad matrices to the smaller inverses. Computational efficiency of both methods can be tested to a large degree using standard linear programming codes.